

09990592 "112301
106211 2650660

| Axis Name | Considered Nodes |
|--------------------|---|
| ancestor | Any node along the path to the root |
| ancestor-or-self | Same, but including the current node |
| attribute | Consider only attribute nodes in the tree |
| child | Any node directly connected to the current node |
| descendant | Any node from the subtree rooted at the current node |
| descendant-or-self | Same, but including the current node |
| following | Any node with id greater than the current node |
| following-sibling | Any same-level node with id greater than the current node |
| parent | The direct predecessor of the current node |
| preceding | Any node with id lower than the current node |
| preceding-sibling | Any same-level node with id lower than the current node |
| self | The current node |

Table 1

09990592.112304

| n | a_i | b_{QHL} | s_{QHL} | t_f |
|-----|-----------------------------|-------------|-----------------------|---|
| n | ancestor | n | ancestors | $oc=XML-$ Element |
| n | ancestor- or-self | n | {ancestor s, base} | $oc=XML-$ Element |
| n | attribute | n | onelevel | $oc=XML-$ Attribute |
| n | child | n | onelevel | $oc=XML-$ Element |
| n | descen- dant | n | subtree | $oc=XML-$ Element |
| n | descen- dant-or- self | n | {subtree, base} | $oc=XML-$ Element |
| n | following | $root(n)$ | subtree | ($\&(oc=XML$ Element) (order> order(n))) |
| n | following -sibling | $parent(n)$ | onelevel | ($\&(oc=XML$ Element) (order> order(n))) |
| n | parent | n | parent | $oc=XML$ Ele ment |
| n | preceding | $root(n)$ | subtree | ($\&(oc=XML$ Element) (order< order(n))) |
| n | preceding -sibling | $parent(n)$ | onelevel | ($\&(oc=XML$ Element) (order< order(n))) |
| n | self | n | base | $oc=XML$ Ele ment |

Table 2

09990592.12301
T0E2T"26506660

| File Name | Size | Apache Cache | Overhead | HLCache | Overhead |
|----------------------------|---------|-----------------|----------|---------|----------|
| mondial- 2.0.XML | 1037629 | 1038094 | 1.00 | 3372502 | 3.25 |
| europa- 2.0.XML | 317913 | 318384 | 1.00 | 1017080 | 3.20 |
| dream. XML | 149524 | 149982 | 1.00 | 303613 | 2.03 |
| SigmoidRe -cord. XML | 494591 | 495056 | 1.00 | 1401088 | 2.83 |
| books1. wml | 3129 | 3586 | 1.15 | 8039 | 2.57 |
| Average | - | - | 1.03 | - | 2.78 |

Table 3

099059.12301
T0E2T"26506660

| File | Nodes/Op | Stor. (s) | Ops/sec. | Retr. (s) | Ops/sec. |
|----------|----------|-----------|----------|-----------|----------|
| Name | s | | | | |
| mondial- | 39633/57 | 13.34 | 2970.99/ | 85.86 | 461.60/6 |
| 2.0.XML | 116 | | 4281.56 | | 65.22 |
| europe- | 12783/18 | 3.88 | 3294.59/ | 26.84 | 476.26/6 |
| 2.0.XML | 186 | | 4687.11 | | 77.57 |
| dream.XM | 3361/623 | 1.19 | 2824.37/ | 10.22 | 328.86/6 |
| L | 1 | | 5236.13 | | 09.69 |
| SigmodRe | 15263/38 | 8.43 | 1810.55/ | 56.33 | 270.95/6 |
| cord.XML | 518 | | 4569.16 | | 83.79 |
| books1.w | 96/138 | 0.0098 | 9795.92/ | 0.18 | 533.33/7 |
| ml | | | 14081.63 | | 66.66 |
| Average | - | - | 2725.12/ | - | 384.27/6 |
| | | | 4693.50 | | 59.07 |

Table 4

0990592-112301
T0E2T1 26506660

| Query | Nr. Result | DOM back-end | HLCaches |
|---|------------|--------------|----------|
| Patterns | Nodes | | |
| /mondial/ country | 260 | 0.69 | 0.05 |
| /mondial// city | 3047 | 217.67 | 11.23 |
| /mondial/ country[@car _code='D'] | 1 | 6.36 | 2.31 |
| /mondial// city[@is_cap ='yes'] | 230 | 276.56 | 17.05 |

Table 5

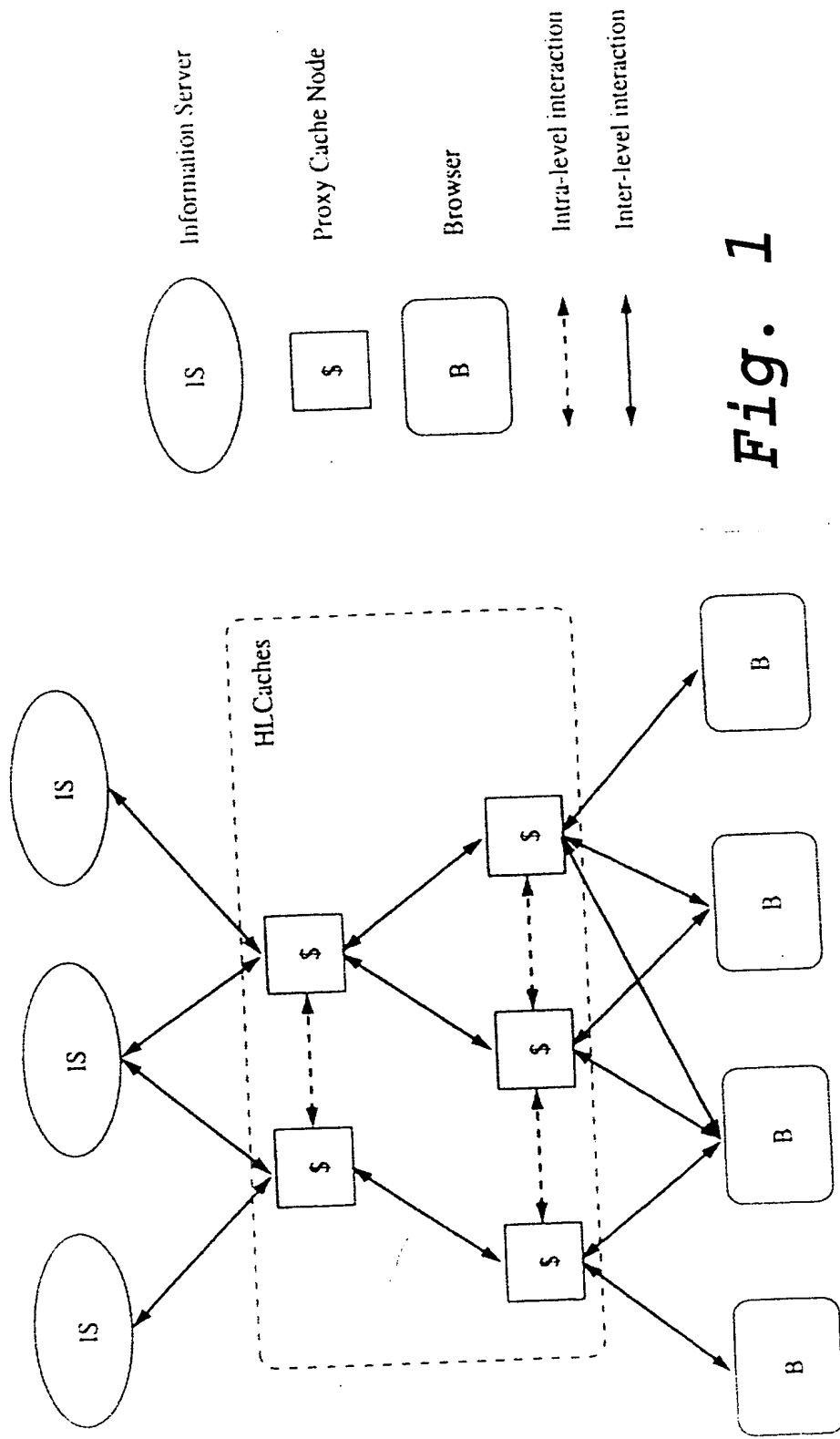
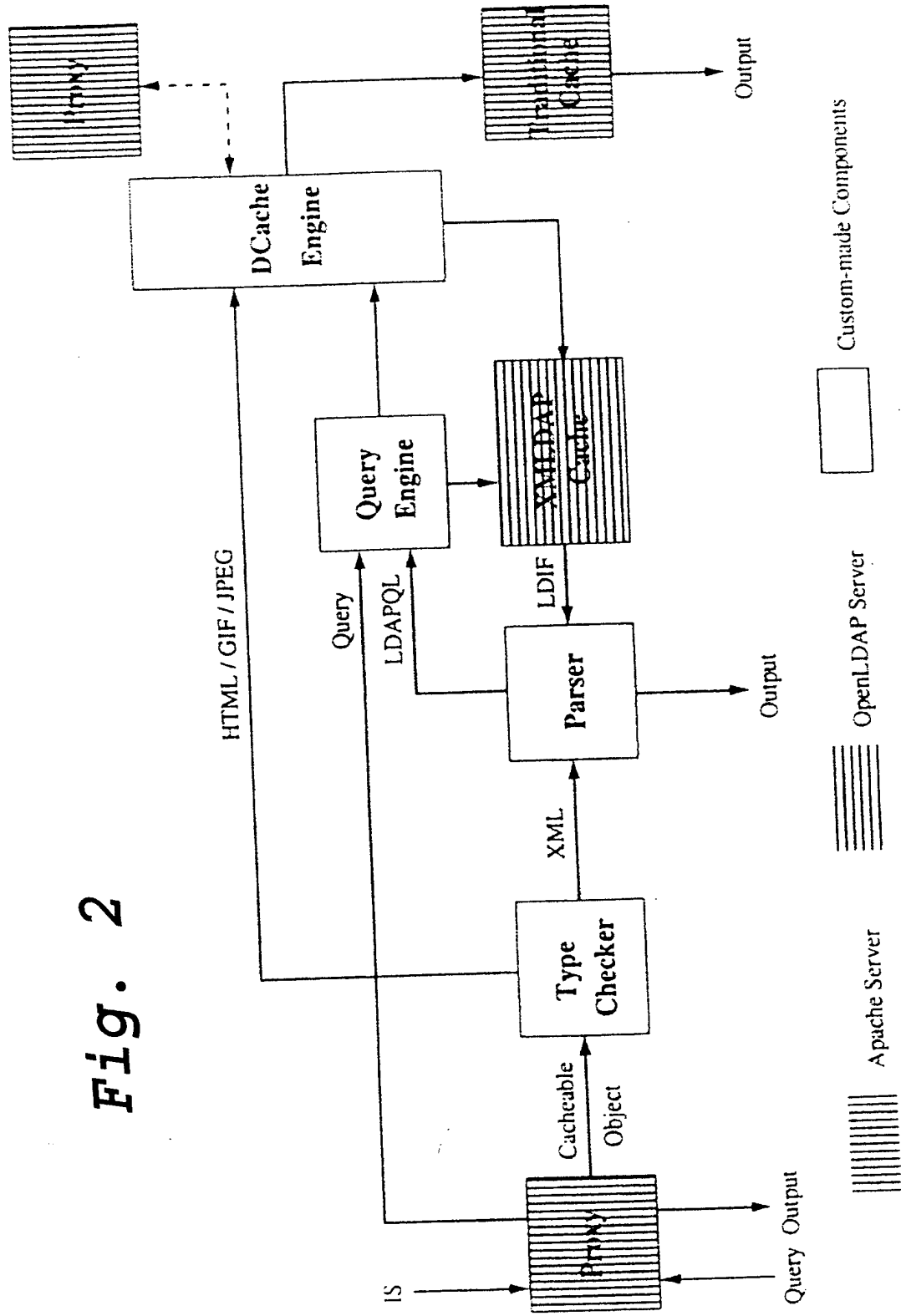


Fig. 1

Fig. 2



09990592.112301
T0421T"26506660

```
XMLNode OBJECT-CLASS ::= {  
    SUBCLASS OF {top}  
    MUST CONTAIN {oc,oid,name}           // required attributes  
    TYPE oc OBJECT-CLASS  
    TYPE oid DN                          // dns formed by oids  
    TYPE name STRING  
}  
  
XMLElement OBJECT-CLASS ::= {  
    SUBCLASS OF {XMLNode}  
    MUST CONTAIN {order}                 // required attributes  
    MAY CONTAIN {value}                  // allowed attributes  
    TYPE order INTEGER  
    TYPE value STRING  
}  
  
XMLAttribute OBJECT-CLASS ::= {  
    SUBCLASS OF {XMLNode}  
    MUST CONTAIN {value}                 // required attributes  
    TYPE value DN, STRING  
}
```

Fig. 3

THE UNIVERSITY OF CHICAGO

```

Algorithm XML2LDAP ( D )

Let D be an XML document to processed from left to right
/* Initialize the current node to the top of the LDAP cache
tree */
CurrentNode = "(cn=Cache,dc=top)"

while there is input i from D
/* If an opening tag is found in the inventive input i */
    if i is
        <tagName attrName0=attrValue0...attrNamen=attrValuen>
            NewNode = XMLElement(tagName)
            link(CurrentNode, NewNode)
            CurrentNode = NewNode
            /* Create the attributes and link them to the
new node */
            for each attrName, attrValue pairs
                NewAttribute = XMLAttribute(attrName,
attrValue)
                link(NewNode, NewAttribute)

            /* If a closing tag is found in the inventive
input i */
            if i is </tagName>
                CurrentNode = Parent(CurrentNode)

            /* else, i is the content of the node */
            else
                CurrentNode.value = i

```

Fig. 4

0990592.112304
T0E2T"26506660

```
<country car_code="D", area="356910",  
    capital="Berlin">  
    <name>Germany</name>  
    <population>83536115</population>  
    <languages percentage="100">  
        German</languages>  
    <province id="B-W", capital="cid-9",  
        country="D">  
        ...  
    </province>  
</country>
```

Fig. 5

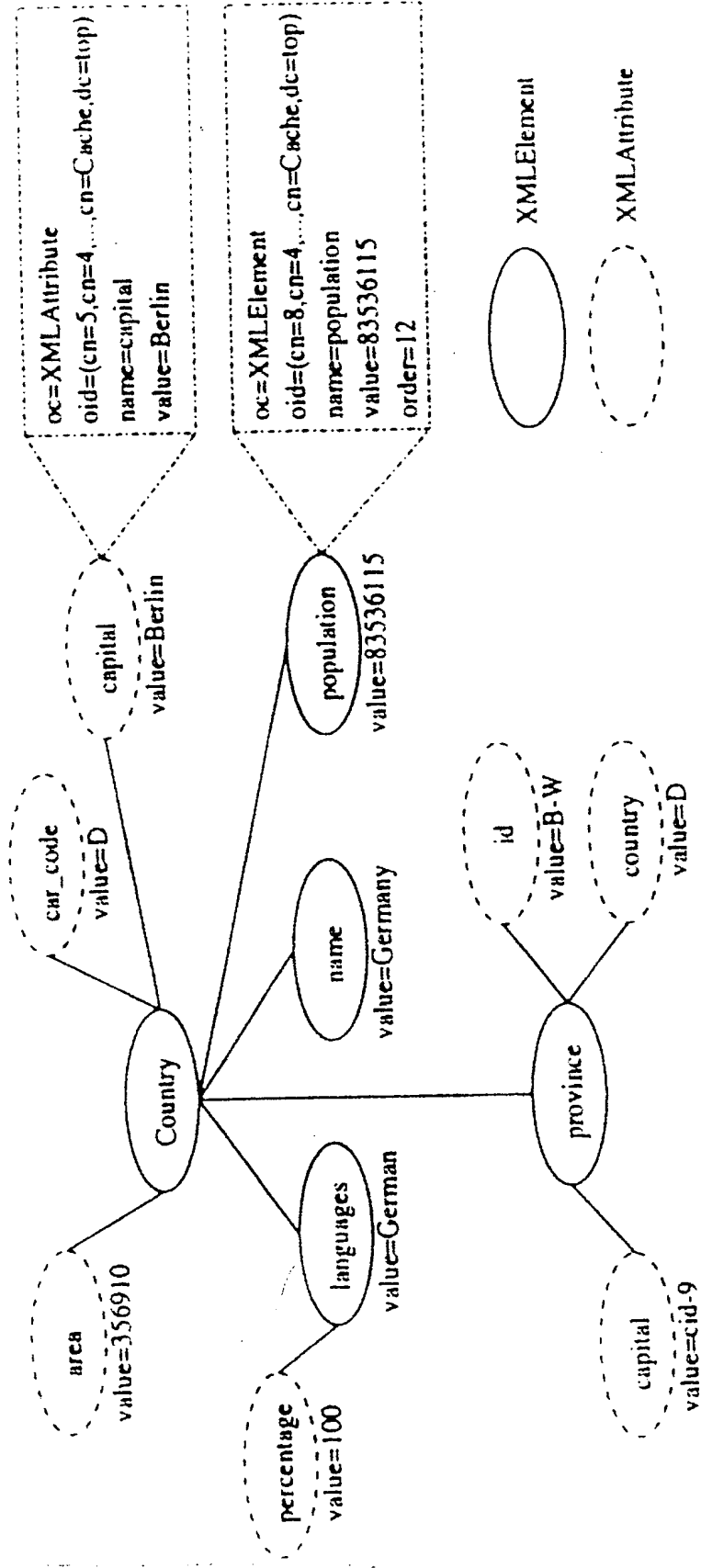


Fig. 6

00000592.12304
T0F2T 2650660

```
<country car_code="D" area="356910" capital="Berlin">
  <name>Germany</name>
  <population>83536115</population>
  <languages percentage="100">German</languages>
  <province id="B-W" capital="cid-9" country="D">
    <name>Baden Wurttemberg</name>
    <area>35742</area>
    <population>10272069</population>
    <city id="cid-9" is_state_cap="yes" country="D"
      province="B-W">
      <name>Stuttgart</name>
      <longitude>9.1</longitude>
      <latitude>48.7</latitude>
      <population year="95">588482</population>
    </city>
  </province>
</country>
```

Fig. 7

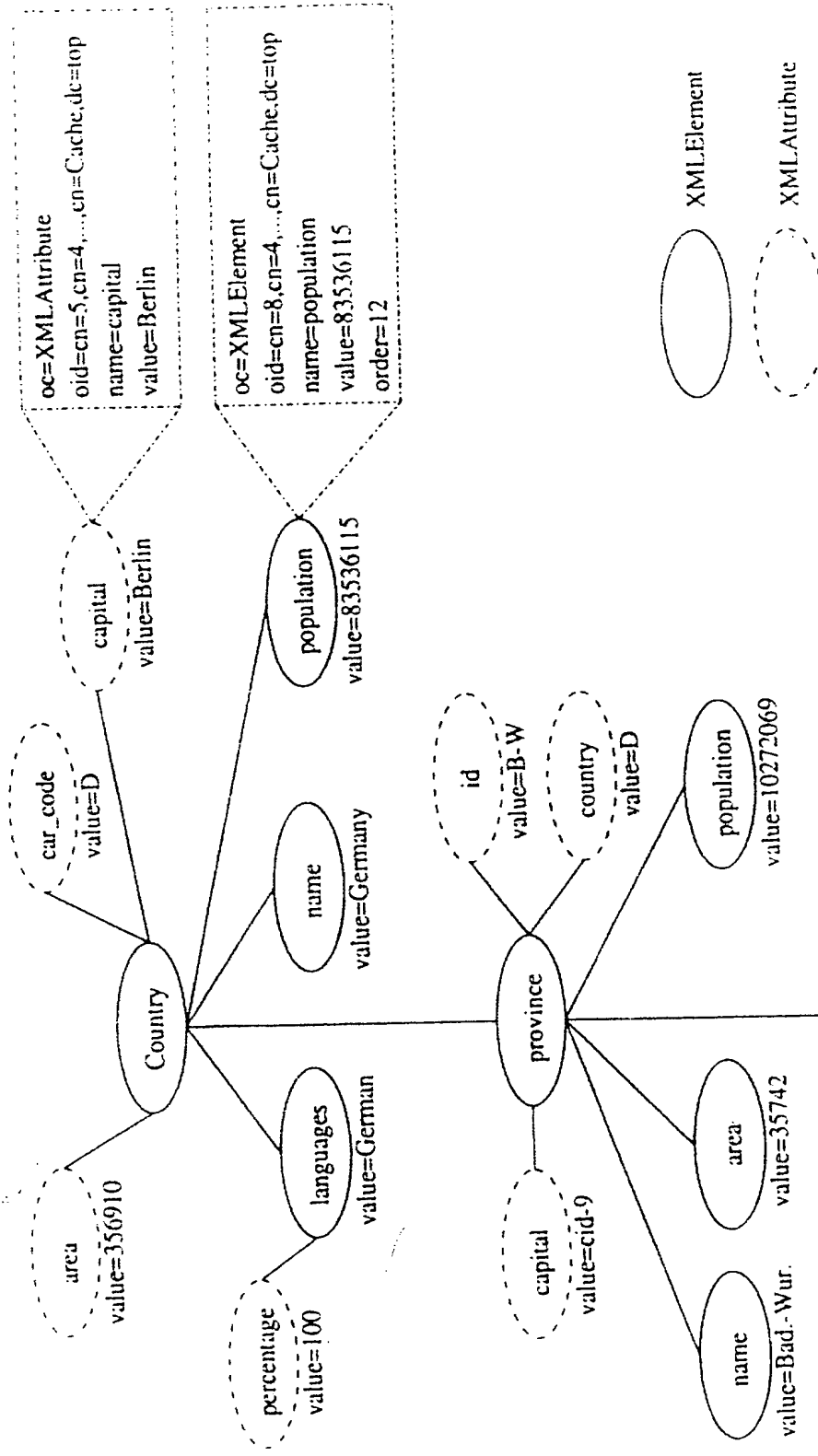


Fig. 8 (part one)

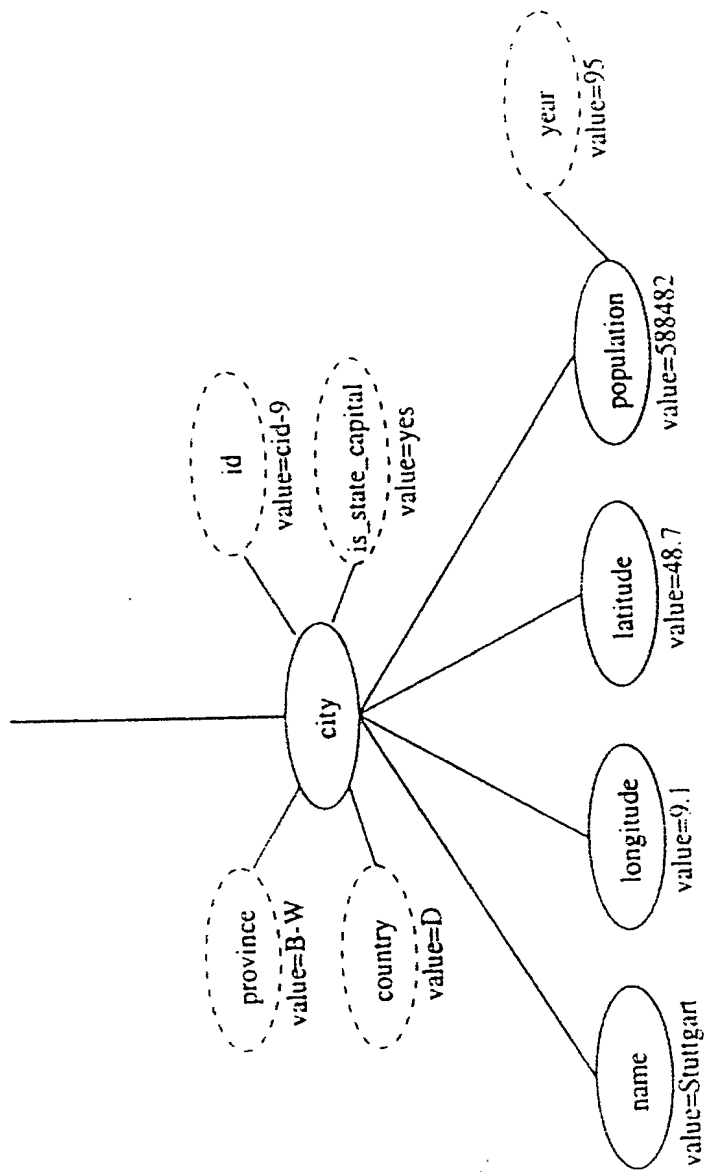


Fig. 8 (part two)

09090592.112301
T052T 2650660

```
XMLQuery OBJECT-CLASS ::=
  SUBCLASS OF top
  MUST CONTAIN oc, hash, context, scope, xpathquery, result,
               create\_time, access\_time, popularity
  TYPE oc OBJECT-CLASS
  TYPE hash STRING
  TYPE context DN
  TYPE scope STRING
  TYPE xpathquery STRING
  TYPE result DN
  TYPE create_time STRING
  TYPE access_time STRING
  TYPE popularity INTEGER
```

Fig. 9

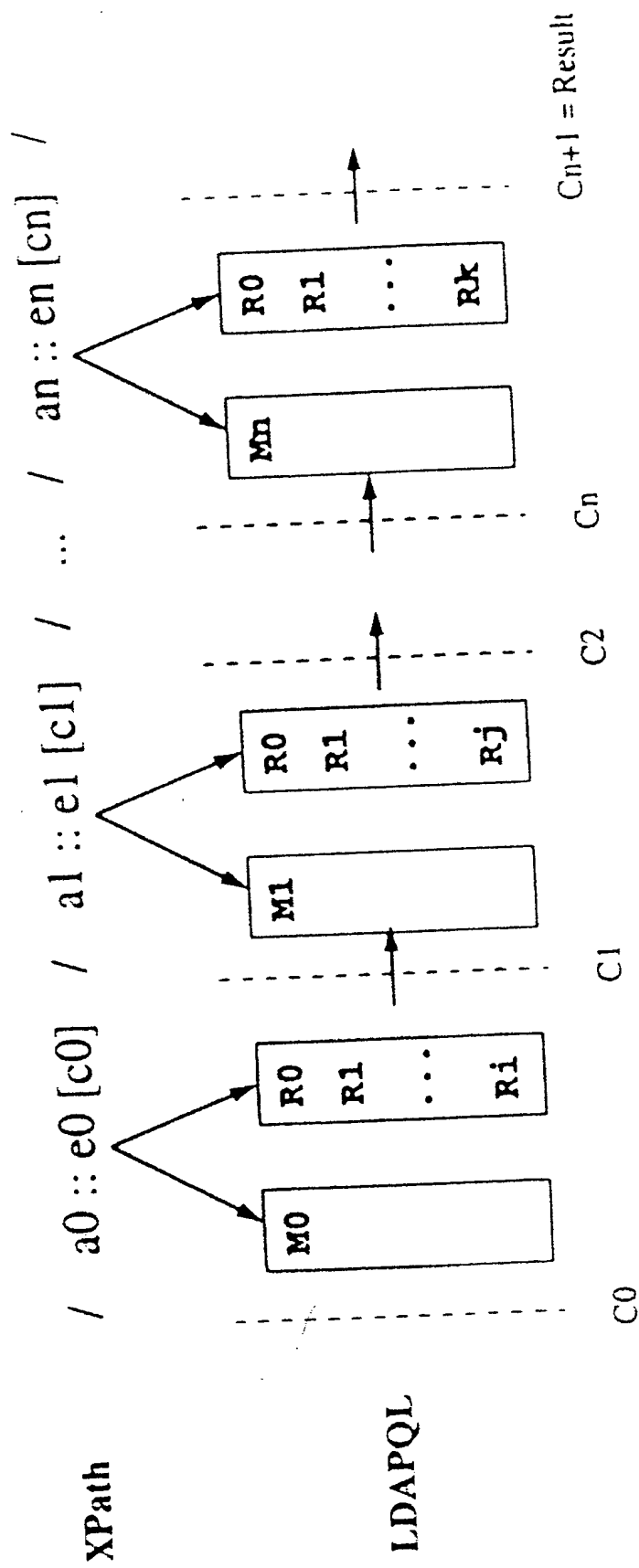


Fig. 10

SECRET

```

/* Initialize C0 to the cache root */

C0 = "cn=Cache,dc=top"


For each subquery qi = (Ci, wi, Ci+1) ∈ QX

    /* Create a new XMLQuery node and initialize its
attributes */

    NewXMLQuery.context = Ci

    NewXMLQuery.xpathquery = wi

    NewXMLQuery.hash = hash( wi )


    /* For each node in the context, evaluate wi on
it */

    for each n ∈ Ci

        Ci+1 = Ci+1 ∪ EVAL( PET( n, wi ) )


    NewXMLQuery.result = Ci+1

```

Fig. 11

Algorithm EVAL (Q, S)

```

/* Q is an LDAPQL query (called main query) */
/* S = {Si} is a set of LDAPQL queries (subordinate)
*/
Result = LDAP( Q )
for each subquery Si ∈ S
    Result = Result ∩ LDAP( Si )
Return Result

```

Algorithm PET(n , w_i)

```

/* n is a distinguished name and wi = ai::ei[ci] */
Let QHL be an LDAPQL query (called main query)
Let S = {Sj} be a set of LDAPQL queries (subordinate)

/* Translate ai into QHL = (bQHL, sQHL, fQHL, pQHL) */
(bQHL, sQHL, fQHL) = BaseScope( n , ai )
for each nodeName ∈ ei
    fQHL = fQHL ∩ (name = nodeName)
    pQHL = {}

/* Translate each predicate cpj into Sj =
(bsj, ssj, fsj, psj) */
Let S = {}
for each cpj ∈ ci
    Let cpj be of the form termj opj valuej
    (bsj, ssj, fsj) = BaseScope(LDAP( QHL ), termj )
    for each (nodeName, nodeValue) ∈ ci
        fsj = fsj ∩ (&(name = nodeName)(value =
nodeValue))
        psj = {}
    S = S ∪ Sj

Return ( QHL , S )

```

Fig. 12

09990592.112301